APPLICATION

FOR

UNITED STATES LETTERS PATENT

TITLE: TRIMMING SURFACES

APPLICANT: DEAN P. MACRI AND KIM PALLISTER

CERTIFICATE OF MAILING BY EXPRESS MAIL

Express Mail Label No. EL584779153US

I hereby certify that this correspondence is being deposited with the United States Postal Service as Express Mail Post Office to Addressee with sufficient postage on the date indicated below and is addressed to the Assistant Commissioner for Patents, Washington, D.C. 20231.

March 31, 2000
Date of Deposit

[signature]
Signature

Derek W. Norwood
Typed or Printed Name of Person Signing Certificate

Docket No.: 10559/154001/P7988

TRIMMING SURFACES

BACKGROUND

This invention relates to trimming a parametric surface.

Trimming is the removal a portion of a surface when rendering an image of the surface. Trimming provides a visual sensation of absence, meaning the appearance of removing a portion of the image.

DESCRIPTION OF DRAWINGS

FIG. 1 is a view of a rendered image.

FIG. 2 is a view of a control point lattice for FIG. 1.

FIG. 3 is a view of a mesh of FIG. 2.

FIG. 4 is a flow diagram of a process for trimming an image according to an embodiment of the invention.

FIG. 5 is a view of a trimming texture.

FIG. 6 is a view of a trimming curve.

FIG. 7 is a view of the trimming texture in FIG. 5 on the mesh in FIG. 3.

FIG. 8 is a view of the trimming texture in FIG. 5 applied to a rendered image of the mesh in FIG. 3.

FIG. 9 is a view of a material texture for the image in FIG. 1.

FIG. 10 is a flow diagram of a process for trimming an image according to another embodiment of the invention.

FIG. 11 is a view of computer hardware used to implement one embodiment of the invention.

FIG. 12 is a view of an alternate embodiment of a trimming texture.

FIG. 13 is a view of the trimming texture in FIG. 11 applied to the mesh in FIG. 3.

Like reference symbols in the various drawings indicate like elements.

DETAILED DESCRIPTION

The following embodiments render two-dimensional (2D) images from data for trimmed three-dimensional (3D) models. A trimmed 3D model is a 3D model that has a portion of the model's surface removed during the rendering process by one or more parametric 2D functions (referred to as 'trimming curves'). The trimmed effect is achieved by applying a 'trimming texture' (that is, a texture map image representation of the trimming curve function or functions) to the surface of the 3D model.

A trimmed image 10 (FIG. 1) of a parametric surface 12 (that is, a surface described with an equation and at least one control points and one or more trimming curves) has a solid portion 14 and a trimmed portion 16. Solid portion 14 provides a visual sensation of being a solid, tangible surface 12. Trimmed portion 16 provides a visual sensation of being absent, missing or transparent. Here, trimmed portion 16 appears blank, but in other embodiments may appear transparent, revealing whatever elements, if any, are located behind surface 12.

Surface 12 of image 10 is modeled using a 3D control point lattice 20 (FIG. 2). Control point lattice 20 has at least one control point, 22a, which defines a location of a point on image 10 in xyz (or other) space. Here, control point lattice 20 has sixteen (16) control points, 22a-p, which together define the location (and hence appearance) of sixteen points on image 10.

Control points 22a-p may be altered to change to the shape (and hence appearance) of image 10. For example, control points 22a-p may be altered to create a trampoline or a balloon effect, where the center portion of image 10 (controlled by 22m-p) rises and falls much like a flexing trampoline or an inflating and deflating balloon. The parametric surface 12 of image 10 may be modeled by one or more mathematical functions. For example, a vector-valued function S(u,v) that maps from a rectangular bounded region in, uv space (the "range" of S) to a bounded surface in XYZ space. U and v are orthogonal, and may be thought of as horizontal and vertical (though u and v may not map to horizontal and vertical in XYZ space). Additionally, there may be one or more trimming functions (trimming curves) that can be used to further reduce the range of S(u,v) and therefore further reduce the surface in XYZ space.

To control the overall appearance of surface 12 between control points 22a-p of control point lattice 20, a mesh 30 (FIG. 3) is obtained. Mesh 30 is made of many polygons 32, here triangles; however other types of polygons may be used. Each polygon 32 includes coordinates for vertices 34a, 34b and 34c that exist in xyz space. These vertices 34 define a face 36 and edges 38a, 38b and 38c for each polygon 32. Mesh 30 serves as an approximation of surface 12. Here, mesh 30 provides a smooth transition of surface 12 between control points 22a-p; however, other types of transitions may be maintained.

FIG. 4 shows a process 40 for trimming surface 12 of image 10. Process 40 trims surface 12 based on a trimming texture 50 (FIG. 5). This trimming texture is typically obtained once per mesh 30 in a "pre-rendering" process and stored prior to rendering the image (a pre-rendering process occurs at a time prior to actually rendering the image). Trimming texture 50 includes a rendered section 52 and a trimmed section 54. Rendered section 52, here shown as white, corresponds to the solid portion 14 of surface 12. Trimmed section 54, here shown as black, corresponds to the trimmed portion 16 of surface 12.

Process 40 obtains (401) a trimming texture 50 based on the trimming curves for surface 12. Trimming curves are defined as curves located in the uv space of mapping function S of a surface, for example surface 12, which restrict the drawing of (that is trims) a portion of the surface 12.

A trimming curve 60 (FIG. 6) may be defined so that either the surface to the left, or the right, of the curve is trimmed. For example trimming curve 60, here a clockwise circle, will create a circle if the surface to the right is trimmed and a hole (such as one shown in FIG. 1) if the surface to the left is trimmed. Trimming curves 60 may also be described as vector-valued functions $C_1(t)$, $C_2(t)$ …, $C_n(t)$.

Process 40 applies (403) trimming texture 50 to mesh 30. Process 40 typically applies trimming texture 50 to mesh 30 in a "run-time" process as a texture map (a run time process occurring at the time of actually rendering the image). FIG. 7 shows trimming texture 50 applied (403) to corresponding vertices 34 of mesh 30 (prior to rendering image 10). Where appropriate, trimming texture 50 changes shape corresponding to the changes in shape (such as flexing or stretching), if any, of mesh 30. This change in shape of trimming texture 50 is a result of the application of texture 50 to mesh 30 and does not require a re-computation of trimming curve 60.

Once applied, process 40 renders (405) the image using trimming texture. Trimming texture 50 affects rendering of the image whenever trimming is desired (for example, by a user).

Specifically, process 40 limits the drawing of surface 12 to solid portion 14, corresponding to rendered section 52 of trimming texture 50. In other embodiments, trimming texture 50 prevents the drawing of pixels in trimmed section 16, corresponding to trimmed section 54. For example, FIG. 8 shows a rendered image 80 of trimming texture 50 applied to mesh 30. As shown, vertices 34 and edges 38 of polygons 32 are allocated (and hence drawn) for rendered portions 82 of mesh 30 corresponding to rendered region 52. Likewise, vertices 34 and edges 38 of polygons 32 corresponding to trimmed region 54 are absent from trimmed portions 84 of mesh 30.

Furthermore, to render image 10 (FIG. 1) a material texture 90 (FIG. 9) may be applied when rendering an image of the surface (405). Material texture 90, here a checkerboard pattern, is applied as a texture map onto mesh 30, however, texture 90 is only rendered in sold regions 14 corresponding to rendered section 52 of trimming texture 50.

FIG. 10 shows an alternative embodiment, process 100, of the invention. Process 100 obtains a surface (1001), for example surface 12, to be rendered. Surface 12 may be obtained by based on a vector-valued function, S(u,v), which generates the surface in XYZ (or other) space.

Process 100 also obtains a trimming curve (1003), or in other embodiments trimming curves, for surface 12. Trimming curves may be obtained (1003) based on vector-valued functions $C_1(t)$, $C_2(t)$ …, $C_n(t)$ which generate trimming curves for surface 12 (in uv space).

Each trimming curve, $C_1(t)$, $C_2(t)$ …, $C_n(t)$ obtained (1003) is mapped (1005) onto a trimming texture 50 for surface 12. The trimming texture 50 is typically obtained once up front in a pre-rendering process based on trimming curve 60 to generate a rendered section 52 and a trimmed section 54 for trimming texture 50.

Process 100 may also obtain (1007) a material texture for surface 12. Material texture 90 (FIG. 9) is, here, a checkerboard material but may be any material provided (for example to a user). Material texture is also typically obtained in a pre-rendering process.

Process 100 obtain (1009) polygons (for example polygons 32) to generate mesh 30 to approximate surface 12. Process 100 may obtain polygons with a tessellation algorithm for approximating surface 12 of image 10 to a desired resolution (for example to support cartoon and/or photo realistic quality images). Typically, process 100 obtains mesh 30 at run-time.

Process 100 renders an image 10 (1011) of surface 12 by applying trimming texture 50 and material texture 90 to polygons 32 of mesh 30. Material texture 90 is drawn as indicated by rendered section 52 and is not drawn as indicated by trimmed section 54. An example of a pseudo-code to implement process 100 is shown in attached appendix.

Processes 40 and 100 may be used to create animation cels for cartooning. For example, a 3D model, such as image 10, may be generated and then positioned in a desired manner in a scene. Processes 40 and 100 may be executed on the model to produce a 2D image for that position. Then, the 3D model can be re-positioned (for example, rotated), and process 40 or 100 (or subparts of 40 and 100) executed on the re-positioned model to produce a 2D image for the new position. This process may be repeated to produce images for any number of positions. In this way, processes 40 and 100 can generate cels automatically.

Processes 40 and 100 may be performed in 3D space, meaning that they are performed using 3D data. By processing data in 3D space (as opposed to 2D space), processes 40 and 100 can be performed using a processor and rendering can be done using graphics accelerator cards of a computer.

FIG. 11 shows a computer 1180 for rendering images using processes 40 and 100. Computer 1180 includes a processor 1181, a memory 1182, a storage medium 1184 (e.g., a hard disk), and a 3D graphics accelerator card 1185 having an alpha-channel for applying textures to 3D model and manipulating 3D data. Storage medium 1184 stores texture maps 1186 containing the trimming texture 50 and material texture 90 described above, 3D data 1189, and computer instructions 1190 for rendering 2D images via processes 40 and 100.

Processes 40 and 100, however, are not limited to use with any particular hardware or software configuration; they may find applicability in any computing or processing environment. Processes 40 and 100 may be implemented in hardware, software, or a combination of the two. Processes 40 and 100 may be implemented in computer programs executing on programmable computers that each include a processor, a storage medium readable by the processor (including volatile and non-volatile memory and/or storage elements), at least one input device, and one or more output devices. Program code may be applied to data entered using an input device to perform processes 40 and 100 and to generate output information. The output information may be applied to one or more output devices, such as display screen 1192.

Each such program may be implemented in a high level procedural or object-oriented programming language to communicate with a computer system. However, the programs can be implemented in assembly or machine language. The language may be a compiled or an interpreted language.

Each computer program may be stored on a storage medium or device (e.g., CD-ROM, hard disk, or magnetic diskette) that is readable by a general or special purpose programmable computer for configuring and operating the computer when the storage medium or device is read by the computer to perform processes 40 and 100. Processes 40 and 100 may also be implemented as a computer-readable storage medium, configured with a computer program, where, upon execution, instructions in the computer program cause the computer to operate in accordance with processes 40 and 100.

A number of embodiments of the invention have been described. Nevertheless, it will be understood that various modifications may be made without departing from the spirit and scope of the invention. For example, trimmed portion 16 may change in shape, size and/or other qualities as desired (for example, as directed by a user or a program). Here, trimmed portion 16 is circular in shape. FIG. 12 shows a trimming texture 1210 having an irregular shape and FIG. 13 shows a rendered image of trimming texture 1210 applied to a mesh 1310. Moreover, trimming texture 50 may further be specifically designed to use texture functionality provided by particular 3D graphic processors. Also, trimming texture 50 may be designed to operate with the alpha-channel of texture blending/mapping hardware in a specific 3D graphics processor. Accordingly, other embodiments are within the scope of the following claims.